Special Feature: Spatial Information Technology towards Intelligent Vehicle Systems

Research Report

Layered Vehicle Control Architecture Coordinated between Multiple Edge Servers

Kengo Sasaki, Naoya Suzuki, Satoshi Makido and Akihiro Nakao

Report received on Jan. 22, 2018

ABSTRACT Recently, various studies have considered the development of autonomous vehicles. Two of the main challenges in developing autonomous vehicles are limitations with sensing and deadlock. To overcome these problems, a cloud-based approach has been proposed. However, the cloud-based approach introduces other problems caused by fluctuations in the Internet traffic. Mobile Edge Computing (MEC), which will be one of the core technologies in the next generation of mobile communication systems, is installed as computational nodes in the mobile network and can alleviate the problems associated with the cloud-based approach.

In this paper, we proposed a layered arrangement of the edge servers as the architecture for the remote vehicle control system. The proposed system is composed of vehicles and multiple layered edge servers, referred to as the "Lower Edge Server (LoES)" and the "Upper Edge Server (UpES)". The LoES is responsible for stable vehicle control during network fluctuations, and the UpES controls the vehicle and optimizes the traffic flow with the broader perspectves by using the sensor information from multiple vehicles. To complement the both edge servers, the LoES monitors the delay between the LoES and UpES, and if the delay is small enough, the UpES controls the vehicles. Otherwise, the LoES controls the vehicles.

KEYWORDS MEC, Edge Server, 5G

1. Introduction

Recently, various studies have considered the development of autonomous vehicles.[1,2] These vehicles determine their trajectory and control themselves by using input from the on-board sensors, such as Lidar, cameras, and GPS. Although the use of such self-driving vehicles is attractive, there are two main problems. The first problem is the limitations of the sensors. Because they cannot see around obstacles, they usually need to stop on a road with poor visibility. The second problem is negotiation; if two vehicles are at an intersection, they cannot decide which vehicle should pass first, and this leads to a deadlock.

To overcome these problems, a cloud-based approach has been proposed.[3,4] These systems periodically aggregate and analyze the sensor information from multiple vehicles in the service area, and provide better control so as to avoid deadlock. Moreover, they can optimize the flow of traffic to avoid congestion. However, the cloud-based approach introduces other problems due to network instability, especially when the control server is deployed on the Internet. How to compensate for this delay is a key problem to solve.

Mobile Edge Computing (MEC)[5] technology has recently attracted attention as one of the core technologies[6] for 5G, the 5th generation mobile communication system, for minimizing network delay. If MEC technology is installed as a computational node in the mobile network, referred to as an "edge server", it can execute all of the processes required for autonomous driving including aggregation, analysis, and control in a local area. In Refs. (7) and (8), an MEC-based approach is proposed.

In this paper, we propose a layered arrangement of the edge servers as architecture for the remote vehicle control system[9,10] as shown in **Fig. 1**. The left column shows the physical network topology, the center column shows the computational node location, and the right column shows the vehicle control function. The proposed system is composed of vehicles and multiple layers of edge servers, referred to as the "Lower Edge Server (LoES)" and the "Upper Edge Server (UpES)".

The LoES is responsible for maintaining stable vehicle control during network fluctuations, and the UpES controls the vehicles with its best effort. In addition to acting as a remote controller, the UpES optimizes the traffic flow with the broader perspective by using sensor information from multiple vehicles. To use both edge servers effectively, the LoES monitors the delay between the LoES and the UpES, and if the delay is small enough, the UpES controls the vehicle. Otherwise, the LoES controls the vehicles.

Because the LoES and the UpES are deployed at different locations in the network, they have different characteristics regarding the number of the connected vehicles, network delays, and computational resources. The LoES is deployed at the base station because network delays should be minimized for stable control. The UpES is deployed at a higher layer so that it can aggregate sensor information across a broader area. There are several candidate locations for the UpES, such as a central office, gateway, and cloud on the Internet.

This paper is organized as follows. In Sec. 2, we show the evaluation platform, which evaluates the effect of network fluctuations on the vehicle control performance. In Sec. 3, we define the network model which is based on measurements of Internet delays. We show the control performance in Sec. 4, and discuss the location of the edge server with some simple calculations in Sec. 5. In Sec. 6 we conclude the paper.

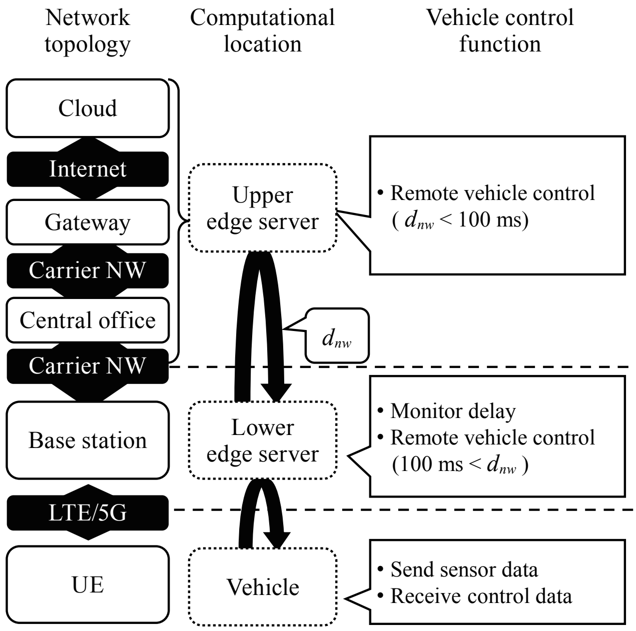

Fig. 1 Proposed system architecture.

2. Evaluation Platform

To evaluate the effect of network fluctuations on the vehicle control performance, we construct a prototype system using a micro-car. **Figure 2** shows the system model for the evaluation prototype. The prototype system consists of micro-cars, Wi-Fi communication system, LoES, UpES, a WAN emulator, and an ultrasonic location system.

2. 1 Vehicles

Radio controllable 1/10-scale micro-cars are used to simulate the vehicles in the system. Each vehicle periodically sends a Vehicle Sensor Data (VSD) packet which includes the vehicle ID, position, velocity, and heading. To imitate the VSD transmission we use an ultrasonic location system, which has a sensing accuracy of about several centimeters.

On the reverse link, the vehicle receives a Vehicle Control Data (VCD) packet, which is the control command including the target vehicle ID, steering angle, accelerator, and brake. The vehicle does not have any control unit inside, and it is fully controlled by the edge servers.

2. 2 Edge Servers

When the LoES receives VSD packets from vehicles, it needs to decide whether to control the vehicles or not. The LoES monitors the delay between the LoES and UpES continuously, and if the average delay d_{ave} is less than a certain threshold D_{th}, the LoES delegates the control to the UpES, as shown by the blue solid line in Fig. 2. In this case, the LoES forwards the VSD packets to the UpES and the VCD packets from the UpES to the vehicles. On the other hand, when $d_{ave} > D_{th}$, the LoES controls the vehicles directly. It receives the VSD packets and VCD packets from the vehicle, as shown by the dotted red line in Fig. 2. We assume that the LoES is deployed at the base station and it can execute the vehicle control without any harmful effects from the network. In other words, the communication delay between the LoES and vehicles is small.

When the UpES is controlling the vehicle, the

UpES receives the VSD packets and sends the VCD packets to the LoES. We assume that the UpES is deployed on a higher layer of the network than the LoES. Although it is true that the UpES can aggregate sensor information from multiple vehicles, it is greatly affected by network conditions such as delays, packet loss, and congestion. Furthermore, because the number of vehicles connected to the UpES is huge, how to manage the computational resources is a big problem for the UpES.

We use the FLARE programmable network router developed by the University of Tokyo[11] for the edge servers in the prototype. FLARE has the capability to not only conduct standard packet processing operations such as routing and QoS scheduling but also conduct versatile processing operations using many core processors. FLARE can execute the functions programmed in the Click Modular Router[12] (Click) programming model for building flexible configurable routers. We implemented the PID control algorithm[13] in Click.

2. 3 Communication Network

Because the impact of the wireless delay on the performance of the vehicle control is considered to be small, we use Wi-Fi for the radio interface. On the contrary, the Internet causes significant performance degradation. The WAN emulator has an important role in the prototype system. We deploy WANem[14] between the LoES and UpES to imitate fluctuations in the network, such as delays and packet loss. For the practical evaluation, we applied the measured delay of the Internet to WANem. The details of the network model are explained in the next section.

3. Network Model

In this section, we explain the network model assumed in this paper. Sec. 3. 1 defines the network topology model to discuss the tradeoff problem between the edge server capability of the users and the network delay. In Sec. 3. 2, we define the delay model

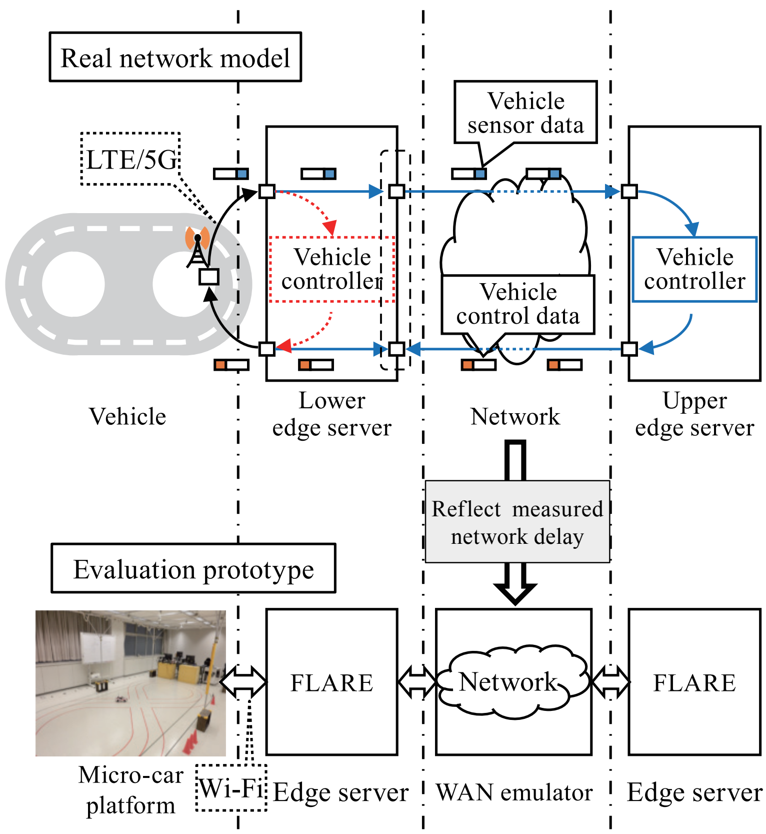

Fig. 2 Evaluation platform.

to evaluate the effect of the deployment location of the edge servers.

3. 1 Topology Model

Figure 3 shows the network topology model. We assume that the network is composed of five elements, which correspond to the location layers, vehicles, base stations, central offices, gateways, and clouds, in order of the lowest to highest layer.

To estimate the magnitude of the number of connected vehicles in a network we consider some relevant statistics. The number of registered vehicles in Japan is 81311679 (as of August 2016).(15) If we assume the percentage of active vehicles at a given time to be 4.2%, the total number of active vehicles N_v is calculated as N_v = 3415090. The total number of base stations in Japan N_{bs} is 51076 (as of September 2016).(16) The number of central offices N_{co} is estimated as 5640 from data for the United States.(17) The number of gateways N_{co} is estimated from the number of Mobile Network Operators (MNOs), and Mobile Virtual Network Operators (MVNOs)(18) in Japan. We assume that three major MNOs have ten gateways each, and the MVNOs have one gateway each, giving N_{gw} = 55. Finally, we assume that there is only one cloud server in the whole network. These assumptions are summarized in **Table 1**.

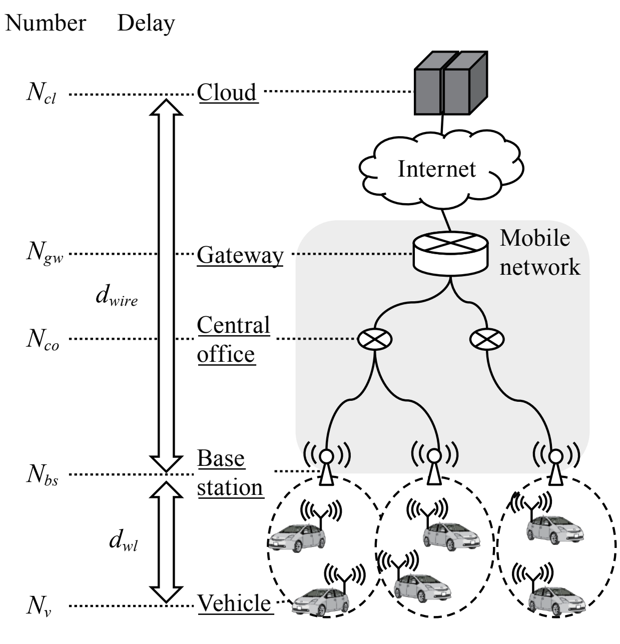

Fig. 3 Assumed network topology.

3. 2 Delay Model

The total network delay can be modeled as the sum of the wireless delay d_{wl} and the wired delay d_w. We assume that the LoES is deployed at the base station and d_{wl} is a small constant. For 5G networks, the requirement of the wireless delay d_{wl} is less than several milliseconds. On the other hand, the wired delay d_w is highly dependent on the location of the UpES. To model the wired delay, we introduce an additional parameter S_r as the suppression ratio of the wired delay. The suppression ratio S_r is defined as the proportional constant of the wired delay to the measurement. If we denote the measured delay at time t as $d_{mod}(t)$, the modeled delay at time t is expressed as

$$d_{mod}(t) = d_{wl} + S_r d_w(t) = d_{wl} + S_r\left(d_{mes}(t) - d_{wl}\right), \qquad (1)$$

where $d_{mes}(t)$ is the measured delay at time t.

To estimate the realistic potential of the real network, we measure the Internet delay. By using the 4G-LTE mobile router, the delay is measured as the round-trip time between the local PC and cloud server on the Internet. In the following section, we assume the suppression ratio S_r to be as shown in **Table 2**.

4. Network Delay and Control Performance

In this section, we evaluate the proposed system. In Sec. 4. 1, we measure the actual network delay for the practical evaluation. In Sec. 4. 2, we evaluate the vehicle control stability.

Table 1 Assumed number.

	Number of elements		Number of connected vehicles	
Elements	Parameter	Value	Parameter	Value
Vehicle	N_v	3415090	N_v/N_v	1
Base station	N_{bs}	51076	N_{bs}/N_{bs}	67
Central office	N_{co}	5640	N_{co}/N_{co}	605
Gateway	N_{gw}	55	N_{gw}/N_{gw}	62092
Cloud	N_{cl}	1	N_{cl}/N_{cl}	3415090

4. 1 Measurement of Internet Delay

Before evaluating the vehicle control performance, we measure the actual Internet delay. **Table 3** shows the measurement parameters. We consider the Google Public DNS[19] as a representative cloud server, and choose it for the destination. We send "ping" commands to the destination every 100 ms and measure the round-trip time. The air interface of the measured network is 4G-LTE, which is provided by a MVNO using NTT DOCOMO's infrastructures. The measurements were done continuously over 24 hours on 1 November 2016.

Figure 4 shows the results of the delay measurements. In Fig. 4, the x-axis shows the time of day and the y-axis shows the measured delay. To evaluate the short time fluctuations of the delay, we use a simple moving average with two different window sizes W_{SMA}. The black dots show the instantaneous delay. The red line shows the results for W_{SMA} = 10, and the blue line shows the results for W_{SMA} = 70. The entire average delay and the packet loss ratio were 86.7 ms and 0.3%, respectively.

Table 2 Location of the network element and suppression ratio S_r.

Location of the network element	S_r
Base station	0
Central office	0.25
Gateway	0.5
Cloud	1.0

Table 3 Parameters for measuring the delay.

Parameter	Value
Destination	Google Public DNS (8.8.8.8)
Transmission interval	100 ms
Wireless communication	4G LTE
Measurement period	1/Nov./2016 0:00-24:00
Window size N_{win}	1070

4. 2 Remote Vehicle Control Performance

To evaluate the vehicle control performance, we apply the delay measurement results to the WANem. We use the trajectory error as the evaluation criterion. Here the trajectory error is defined as the distance between the desired path and the observed micro-car trajectory. We assume that the LoES is located at the base station and the UpES is at the Internet cloud, i.e., $S_r = 1$.

Table 4 shows the parameters for the evaluation. In Table 4, the driving time refers to the time the results were recorded, from 13:29 to 13:35. We choose this period because the fluctuation in the delay is large. In addition to the time selection, we set W_{SMA} = 10 so that the server switches between the LoES and the UpES as frequently as possible. Because the vehicle velocity is scaled by a factor of 10, a velocity of 1 m/s corresponds to a velocity of 36 km/h for a real-sized vehicle. The server switching threshold D_{th} comes

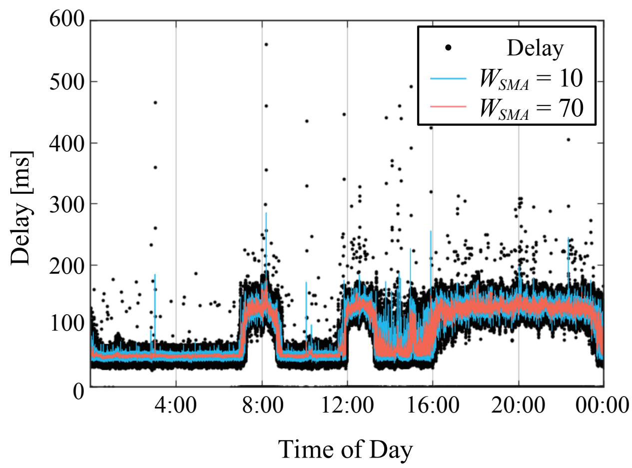

Fig. 4 Monitoring the network delay of MVNO X.

Table 4 Parameters for the measurement of the driving trajectory.

Parameter	Value
Drive time	6 [min]
Velocity	1 [m/s]
Transmission cycle	100 [ms]
Threshold delay D_{th}	100 [ms]
Window size W_{SMA}	10

from the discussion in Ref. (9). The wireless delay d_{wl} is assumed to be constant.

Figure 5 shows an example of the driving trajectory, and **Fig. 6** shows the CDF of the trajectory error. We consider three methods for controlling the vehicles, (I) only the LoES, (II) only the UpES, and (III) the proposed control method. In case (I), the LoES can control the vehicle without the Internet. Therefore, the degradation in performance from (II) to (I) can be seen as the effect of the Internet delay. In both of the above figures, case (III) is almost identical to case (I). This means the server switching algorithm functions correctly. From Fig. 6, the probabilities of a vehicle going outside its lane are 42% for case (II) and 2% for case (III).

The experimental results confirm the effectiveness of the remote vehicle control. We further discuss the proposed control method from the view of computational resource management in the next section.

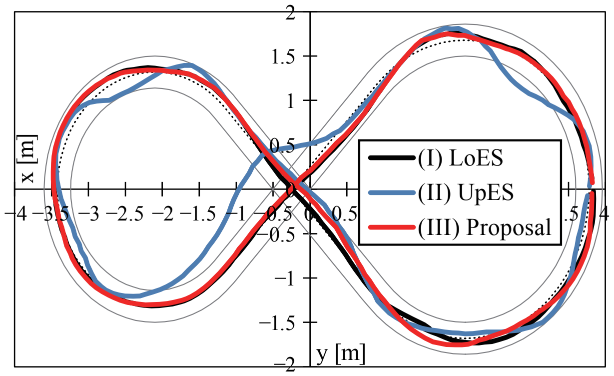

Fig. 5 Example of the driving trajectory.

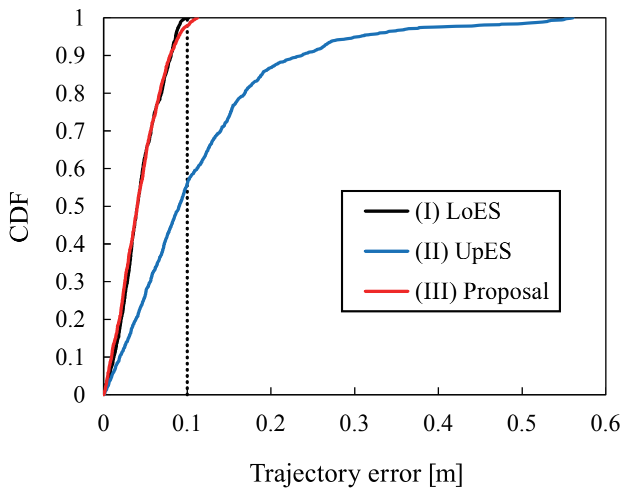

Fig. 6 CDF of the trajectory error.

5. Effect of the Edge Server Location

The proposed layered architecture of the remote vehicle control system is effective. In this section we consider the edge server location from the viewpoint of efficient server utilization.

Consider if the computational resources available for the LoES are very limited and it cannot manage all of the connected vehicles. In this case the UpES needs to compensate for the limited resources of the LoES. In fact, 52% of the control power belongs to the LoES in the case discussed in Sec. 4. To reduce the computation load of the LoES, the UpES should be located near the LoES.

In this section we discuss the location of the UpES with some provisional calculations to validate the relationship among the network delay, the number of vehicles connected to the edge server, and the available computational resources.

5. 1 Control Rate and Number of Vehicles

To evaluate the effect of the UpES location, we apply the network model discussed in Sec. 3. To calculate the control rate of the UpES, which is defined as the fraction of time that the UpES is controlling the vehicle, we use the measured delay shown in Fig. 4 and the parameters in **Table 5**. Most of the evaluation parameters are the same as the values in Table 4. To apply the suppression ratio S_r, the wireless delay d_{wl} is set to 20 ms, which is a large increase from the 5G requirement. To avoid frequent switching between the LoES and the UpES, the delay averaging window size W_{SMA} is set to 70.

Figure 7 shows the control rate of the UpES and

Table 5 Parameters for the simulation of the control rate.

Parameter	Value
Threshold delay D_{th}	100 [ms]
Window size W_{SMA}	70
Wireless network delay d_{wl}	20 [ms]
Suppression ratio S_r	1.0, 0.99, …, 0.0

the number of connected vehicles as a function of the suppression ratio S_r. Figure 7 shows the trade-off between the number of connected vehicles and the computational resources of the UpES. In the case that the UpES is deployed in an Internet cloud, the UpES is expected to be able to accommodate over 3 million vehicles with a control rate of 51%. When the UpES is deployed at the gateway, the control rate increases to 76%. If the UpES is deployed at a lower layer such as a central office or base station, the control rate reaches 100%.

A steep increase in the control rate can be seen around $S_r = 0.5$. This implies that if the Internet delay is halved in the future, we can omit the edge servers. Figure 7 also provides another insight for the UpES location. The UpES location at the gateway is near the branch point of the remote vehicle control. If we deploy the UpES at a higher layer than the gateway, we need to deploy additional LoESs at the lower layer to provide contiguous remote vehicle control services.

Figure 8 shows the control modes of the LoES and the UpES when the UpES is deployed at a cloud (S_r = 1), gateway (S_r = 0.5), and central office (S_r = 0.25). In Fig. 8, the black dots show the instantaneous delays. The blue and red lines show which server is controlling the vehicles.

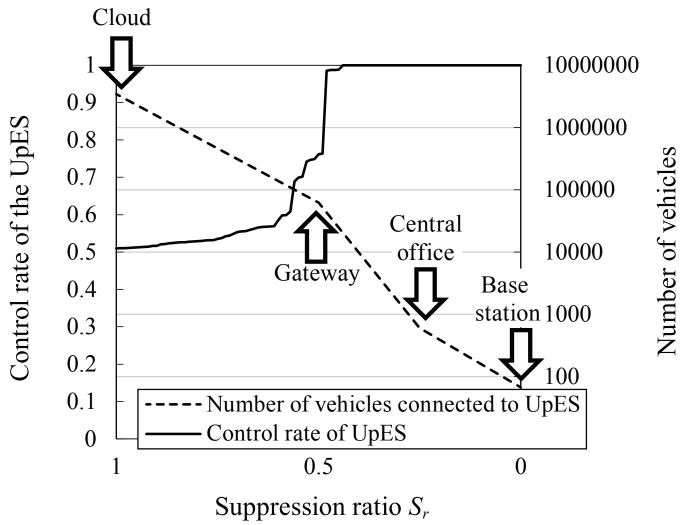

Fig. 7 Control rate of the UpES and the number of connected vehicles.

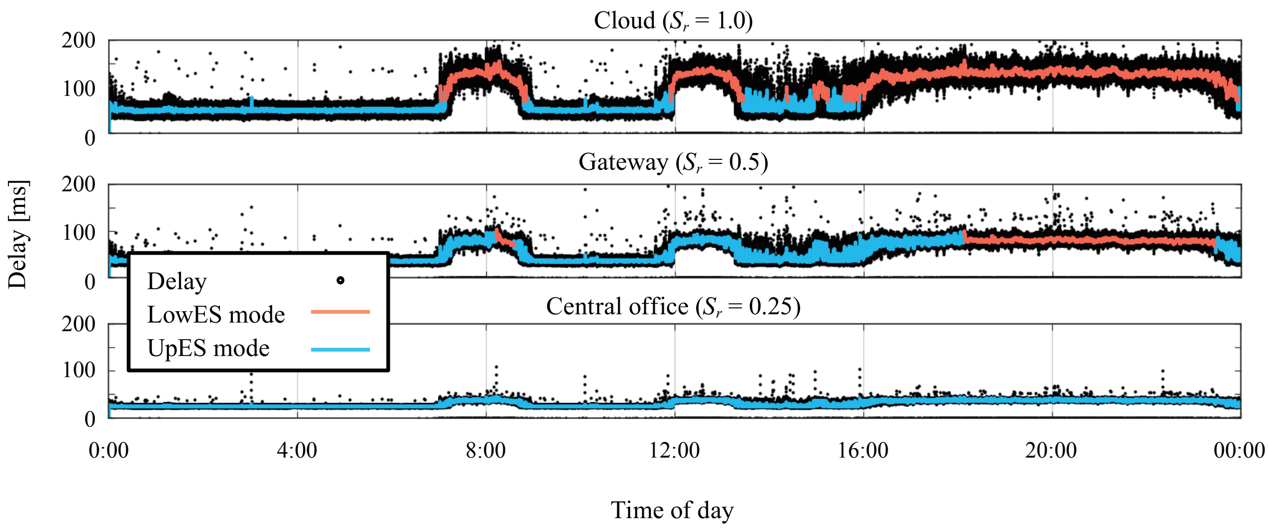

Fig. 8 Entire day control ratio.

6. Conclusion

In this paper, we proposed a layered arrangement of edge servers as the architecture for a remote vehicle control system. The proposed system is composed of vehicles and multiple layered edge servers, referred to as the LoES and UpES. The LoES is responsible for stable vehicle control during network fluctuations, and the UpES controls the vehicles and optimizes the traffic flow with the broader perspective by using sensor information aggregated from multiple vehicles.

From an evaluation using a prototype, we showed that the proposed system can control vehicles without delays due to problems with the Internet. In addition, we discussed the location of the UpES with some provisional calculations and showed the trade-off between the number of connected vehicles and the computational resources required by the UpES.

References

(1) Guizzo, E., “How Google’s Self-driving Car Works”, *IEEE Spectrum Online*, <https://spectrum.ieee.org/automaton/robotics/artificial-intelligence/how-google-self-driving-car-works>, (accessed 2018-01-22).

(2) Broggi, A. et al., “Autonomous Vehicles Control in the VisLab Intercontinental Autonomous Challenge”, *Annu. Reviews in Control*, Vol. 36, No. 1 (2012), pp. 161-171.

(3) Suzuki, N. et al. “A Low Latency Processing Method on Stream-based LDM toward Cloud-based Automated Driving”, *Proc. Embedded System Symp.* (in Japanese) (2015), pp. 84-92, IPSJ.

(4) Kumar, S. et al., “Carspeak: A Content-centric Network for Autonomous Driving”, *ACM SIGCOMM Comput. Commun. Review*, Vol. 42, No. 4 (2012), pp. 259-270.

(5) Hu, Y.-C. et al., “Mobile Edge Computing: A Key Technology Towards 5G”, *ETSI White Paper*, No. 11 (2015), pp. 1-16, ETSI.

(6) Network Architecture Committee, “Networking Technologies for 5G”, *5GMF*, <http://5gmf.jp/wp/wp-content/uploads/2016/08/5GMF_NetArch_Whitepaper_v17.pdf>, (accessed 2018-01-22).

(7) Ito, M. and Ikeda M., “Vehicle Control Using Edge Computing and Mobile Network”, *23rd ITS World Congr.* (2016), ITS.

(8) Nokia, “NOKIA Introduced 5G Autonomous Car”, *AVING GLOBAL NETWORK*, <http://us.aving.net/news/view.php?articleId=1365627>, (accessed 2018-01-22).

(9) Sasaki, K. et al., “Vehicle Control System Coordinated Between Cloud and Mobile Edge Computing”, *Proc. SICE Annu. Conf.* (2016), pp. 1122-1127, SICE.

(10) Sasaki, K. et al., “Layered Vehicle Control System Coordinated Between Multiple Edge Servers”, *Proc. IEEE Netsoft* (2017), pp. 1-5, IEEE.

(11) Nakao, A., “Revisiting ‘Clean-state’ Approach to Re-designing the Internet”, *14th GENI Eng. Conf.* (2012), GENI.

(12) Kohler, E. et al., “The Click Modular Router”, *ACM Trans. Comput. Syst.*, Vol. 18, No. 3 (2000), pp. 263-297.

(13) Araki, M. “PID Control”, *Control Systems, Robotics and Automation: System Analysis and Control: Classical Approaches II Volume 2* (2009), pp. 58-79, Eolss.

(14) Kalitay, H. K. and Nambiarz, M. K., “Designing WANem: A Wide Area Network Emulator Tool”, *Proc. COMSNETS* (2011), pp. 1-4, IEEE.

(15) Automobile Inspection & Registration Information Association, “*Jidousha hoyuu daisuu*” (in Japanese), <http://www.airia.or.jp/publish/statistics/ number.html>, (accessed 2018-01-22).

(16) Ministry of Internal Affairs and Communications, “*Musen kyoku toukei jouhou*”, *The Radio Use Web Site* (in Japanese), <http://www.tele.soumu.go.jp/j/musen/toukei/>, (accessed 2018-01-22).

(17) Peterson, L. et al., “Central Office Re-architected as a Data Center”, *IEEE Communications Magazine*, Vol. 54, No. 10 (2016), pp. 96-101.

(18) Ministry of Internal Affairs and Communications, “*MVNO sābisu no riyou doukou nado ni kansuru dēta no kouhyou*” (in Japanese), <http://www.soumu.go.jp/menu_news/s-news/01kiban02_02000151.html>, (accessed 2018-01-22).

(19) Google, “Google Public DNS”, <https://developers.google.com/speed/public-dns/>, (accessed 2018-01-22).

Kengo Sasaki

Research Fields:
- Vehicle Communication
- Mobile Edge Computing

Academic Societies:
- Information Processing Society of Japan
- The Institute of Electronics, Information and Communication Engineers
- IEEE

Naoya Suzuki

Research Fields:
- Parallel and Distributed Computing
- ITS Platform

Academic Degree: Dr.Eng.

Academic Societies:
- Information Processing Society of Japan
- Association for Computing Machinery

Satoshi Makido

Research Fields:
- ITS Systems
- Wireless Communication Systems
- GNS Positioning

Academic Degree: Dr.Eng.

Academic Societies:
- The Institute of Electronics, Information and Communication Engineers
- Information Processing Society of Japan
- Society of Automotive Engineers of Japan
- IEEE

Akihiro Nakao*

Research Fields:
- Network Virtualization
- Mobile Edge Computing

Academic Degree: Dr.Sci.

Academic Societies:
- Information Processing Society of Japan
- The Institute of Electronics, Information and Communication Engineers
- IEEE
- Association for Computing Machinery

Award:
- ITU-AJ Encouragement Award, 2016

*University of Tokyo

